**MACHINE LEARNING**

**SYSTEMATIC APPROACH**

**FOREST COVER TYPES PREDICTION**

**USING MACHINE LEARNING TECHNIQUES**

**DR DIPANJAN GOSWAMI**

**LEAD INVESTIGATOR- DR DIPANJAN GOSWAMI**

[INTRODUCTION](#_pnvy6uh7dl07)

[1.1](#4d34og8) CARTOGRAPHIC ANALYSIS 4

[1.2](#4d34og8) [AIM OF THE PROJECT](#4d34og8) 4

1.3 PROBLEM STATEMENT 4

[LITERATURE SUMMARY](#_7qaje0copvkt)

2.1. HANDLE IMBALANCED CLASSIFICATION PROBLEMS IN MACHINE LEARNING 5

[DATA DESCRIPTION](#_i878u25o9wcv)

3.1 DATASET……………………………………………………………………………………………………………….........6

3.2 VARIABLES CONSIDERED FOR ANALYSIS…………………………………………………………………………………....6

[3.3](#3rdcrjn) TARGET VARIABLES………………………………………………………………………………………………………....6

[EXPLORATORY DATA ANALYSIS](#_chqhiae1qupt)

4.1 INTRODUCTION 7

4.2 COMPARISON OF FOREST-COVER-TYPE FOR ALL ATTRIBUTES USING PAIR PLOT 7

4.3 HEAT MAP OF ATTRIBUTE CORRELATION 8

4.4 PERCENTAGE DISTRIBUTION OF 4 CLASSES OF FOREST-COVER-TYPE 9

4.5 SKEWNESS PLOT FOR VARIOUS PREDICTORS/COLUMNS 9

4.6 SCATTERPLOT OF HILLSHADE INDEXES 10

4.7 STATISTICAL ANALYSIS 11

DATA CLEANING

5.1 TRANSFORMATIONS..…………………………………………………………………………………………………… 15

5.2 OUTLIER TREATMENT…………………………………………………………………………………………………… 16

[INITIAL APPROACH](#_mgl6a11ho1qu)………………………………………………………………………………………….. 17

[SENSITIVITY ANALYSIS](#_9gwznwmmq8ls)………………………………………………………………………..…… 18

[APPENDIX](#_5yt80s5wvd9f) …………………………………………………………………….…………………… 19

8.1 BASE MODEL’S CODE 19

8.2 FUNCTION USED FOR MODELING 21

8.3 MODELING AFTER DROPPING SOIL TYPES 22

8.4 FUNCTION THAT USES MODELIS 23

8.5 FUNCTION TO OUTPUT RANDOM FOREST FEATURE IMPORTANCE 24

8.6 MODEL WITH FE SHORTEST DISTANCE TO HYDROLOGY 24

8.7MODEL WITH FE HILLSHADE INDEX 25

8.8 MODEL WITH SIN-COS OF SLOPE 26

8.9 MODEL WITH ADDING-SUBTRACTING ALL COMBINATIONS OF HORIZONTAL DISTANCES 27

8.10 AFTER APPLYING LOG-TRANSFORM ON ELEVATION 28

8.11 AFTER APPLYING PD-QCUT ON ELEVATION 29

# INTRODUCTION

### 1.1 CARTOGRAPHIC ANALYSIS

The data is conducted in the Roosevelt National Forest, located in the Front Range of northern Colorado determined from US Forest Service (USFS) Region 2 Resource Information System (RIS) data. We have chosen to study various cartographic factors given in our dataset to identify the forest cover type among the 7 cover types present in Colorado type classes used in this study were lodgepole pine (Pinus contorta), spruce/fir (Picea engelmannii and Abies lasiocarpa), ponderosa pine (Pinus ponderosa), Douglas-fir (Pseudotsuga menziesii), aspen (Pop-ulus tremuloides), cottonwood/willow (Populus angustifolia, Populus deltoides, Salix bebbiana, Salix amygdaloides), and krummholz. It is observed that in our data we have 4 types of Wilderness Areas which are as follows:

* [Cache La Poudre Wilderness](https://en.wikipedia.org/wiki/Cache_La_Poudre_Wilderness), 14.47 square miles
* [Comanche Peak Wilderness](https://en.wikipedia.org/wiki/Comanche_Peak_Wilderness), 104.4 square miles
* [Neota Wilderness](https://en.wikipedia.org/wiki/Neota_Wilderness), 15.51 square miles (partly in [Routt NF](https://en.wikipedia.org/wiki/Routt_National_Forest))
* Rawah Wilderness, 119.4 square miles (partly in Routt NF)

Generally, cover type data are either directly recorded by field personnel or estimated from remotely sensed data but the data we are having is cartographic data which is usually used to save and manage geographical information through Geographic Information System (GIS). The techniques used otherwise can turn out to be costly or time consuming in certain conditions. Also, it can become economically / legally impossible to collect and manage natural resource inventory. Cartographic methods can help save time and improve the efficiency of the workflow of identifying basic natural resource inventory details such as Forest Cover Type with great accuracy.

### 1.2 AIM OF THE PROJECT

Forest cover type being one of the most basic characteristics recorded in natural resource inventory information. Therefore, the forest cover classification needs to be highly accurate. Since, it is an extremely vital piece of information for the private, state, or federal land management agency.

We aim to predict these forest cover types with a reasonable level of accuracy using given cartographic data and aid in the appropriate natural resource inventory management for federal land management agencies.

### 1.3 PROBLEM STATEMENT

* Prediction of 4 Forest cover types from the aforementioned categories given in the data using Cartographic variables as the target variable to get accurate predictions viable for state, private and federal agencies.
* Which one among Elevation, Slope & Aspect actually turns out to improve predictions in the best possible way?

# LITERATURE SUMMARY

### 2.1. HANDLE IMBALANCED CLASSIFICATION PROBLEMS IN MACHINE LEARNING

While performing the conventional machine learning on the data it is evident to manage the basic problems first that is an imbalanced data-based model will always be inaccurate and biased. Considering this to be a multiclass problem the number of observations in one class or more than one class turns out to be significantly less than other classes. Although, both the minority as well as the majority class seem to have enough observations for them to represent their respective classes in our data. Therefore, the bias will definitely exist but it may come out to have a very minimal effect upon the model predictions. Standard classifier algorithms like Decision Tree and Logistic Regression have a bias towards classes which have larger number of instances. This happens because Machine Learning Algorithms are usually designed to improve accuracy by reducing the error. Thus, they do not take into account the class distribution / proportion or balance of classes. They tend to only predict the majority class data. The features of the minority class turnout to be treated as noise and are often ignored.

Author: Upasana | Consultant of Data & Analytics in KPMG.

<https://www.analyticsvidhya.com/blog/2017/03/imbalanced-classification-problem/>

# DATA DESCRIPTION

### 3.1 DATA SET:

Predicting forest cover type from cartographic variables only (no remotely sensed data). The actual forest cover type for a given observation (30 x 30 meter cell) was determined from US Forest Service (USFS) Region 2 Resource Information System (RIS) data. Independent variables were derived from data originally obtained from US Geological Survey (USGS) and USFS data. Data is in raw form (not scaled) and contains binary (0 or 1) columns of data for qualitative independent variables (wilderness areas and soil types).  
This study area includes four wilderness areas located in the Roosevelt National Forest of northern Colorado. These areas represent forests with minimal human-caused disturbances, so that existing forest cover types are more a result of ecological processes rather than forest management practices.  
Some background information for these four wilderness areas: Neota (area 2) probably has the highest mean elevation value of the 4 wilderness areas. Rawah (area 1) and Comanche Peak (area 3) would have a lower mean elevational value, while Cache la Poudre (area 4) would have the lowest mean elevation value.  
As for primary major tree species in these areas, Neota would have spruce/fir (type 1), while Rawah and Comanche Peak would probably have lodgepole pine (type 2) as their primary species, followed by spruce/fir and aspen (type 5). Cache la Poudre would tend to have Ponderosa pine (type 3), Douglas-fir (type 6), and cottonwood/willow (type 4).  
The Rawah and Comanche Peak areas would tend to be more typical of the overall dataset than either the Neota or Cache la Poudre, due to their assortment of tree species and range of predictive variable values (elevation, etc.) Cache la Poudre would probably be more unique than the others, due to its relatively low elevation range and species composition.

(<https://archive.ics.uci.edu/ml/datasets/covertype>)

### 3.2 VARIABLES CONSIDERED FOR ANALYSIS

The dataset consists of 54 variables and 581012 instances for the prediction of Forest cover types from the 54 variables 1 variable is the target variables.

### 3.3 TARGET VARIABLES:

The target variables are four categories and are characterized by 1, 2, 3, 7 in our data set where ‘1’ represents Spruce/Fir, ‘2’ represents Lodgepole Pine, ‘3’ represents Ponderosa Pine, ‘7’ represents Krummholz.

# EXPLORATORY DATA ANALYSIS

### 4.1 INTRODUCTION

EDA is a general approach to exploring datasets by means of simple summary statistics and graphic visualizations in order to gain a deeper understanding of the data.

### 4.2 COMPARISON OF FOREST-COVER-TYPE FOR ALL ATTRIBUTES USING PAIR PLOT

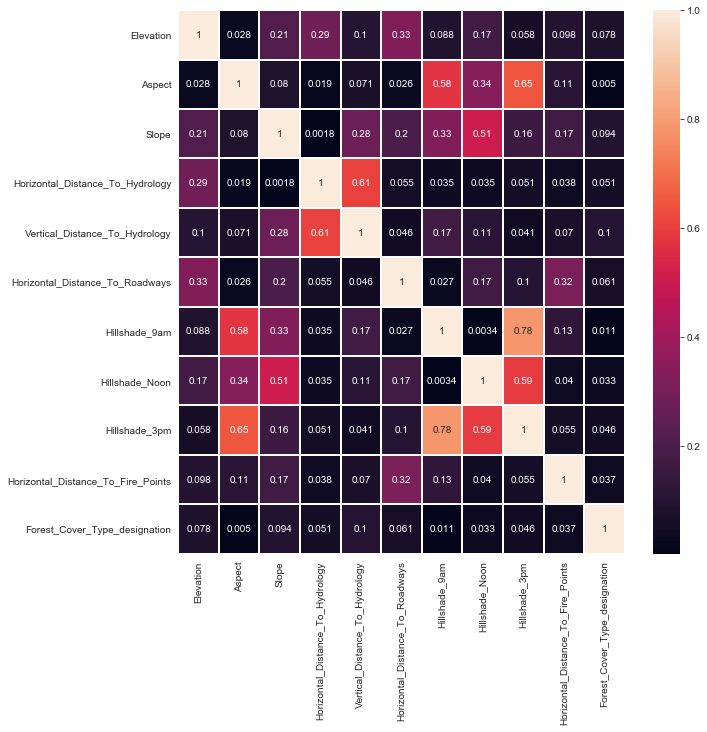


Elevation is the Altitude of the GeoSpatial point that we are trying to measure from earth's surface. In our analysis of the above shown plot, it seems that Elevation turns out to be one of the best features which will help classify the different forest cover types in the best way possible. Since, the multimodal characteristic is distinctly visible. Although, we will be checking all this by performing appropriate statistical tests.

The scatter plots with Elevation and other features like Aspect, Slope and all Hillshade index shows good relation with each other and makes it pretty obvious that an axis parallel decision boundary will show better results than a linear decision boundary. Hence, tree based models should give us better results.

### 4.3 HEATMAP OF ATTRIBUTE CORRELATION

As we have seen from the above plot, heatmap seems to give similar inferences about our target variable relationship with other features. Highest value from the correlation plot for the forest-cover-type is of “Elevation” which shows that it might be considered among one of the best predictors of forest-cover-type, after that we’ve got two other features which have comparatively lesser relation which are “Distance to roadways” and “Slope”.



Also we can see here that there is a strong correlation between “Hillshade\_9am and Hillshade\_3pm”, “Slope and Hillshade\_noon”, “Aspect and Hillshade\_9am”, “Horizontal\_Distance\_to\_Hydrology” and “Vertical\_Distance\_to\_Hydrology” which shows that multicollinearity is present here. Also, these variables can be further explored with its effects upon applying transformations & feature engineering along with the domain knowledge.

### 4.4 PERCENTAGE DISTRIBUTION OF 4 CLASSES OF FOREST-COVER-TYPE

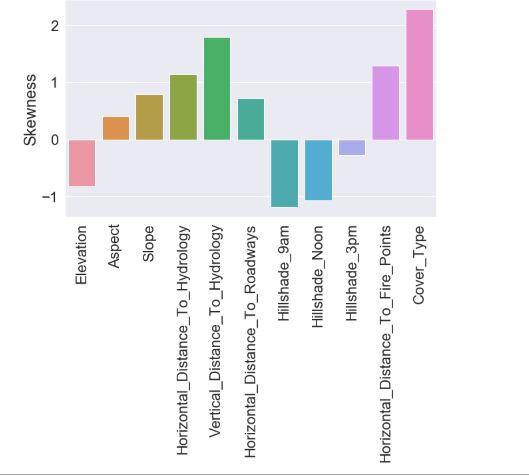


Figure: % Distribution in Target Variables

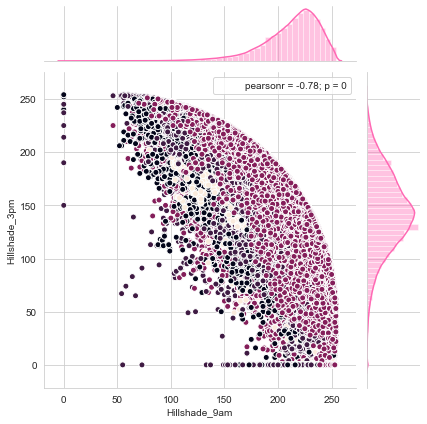
Lodgepole Pine: 48.75%, Spruce/Fir: 36.46%, Ponderosa Pine: 6.15%, Krummholz: 3.53%

### 4.5 SKEWNESS PLOT FOR VARIOUS PREDICTORS/COLUMNS-

As we can see from the plot below, the columns that are positively skewed are: Slope, Aspect, Horizontal\_Distance\_To\_Hydrology, Vertical\_Distance\_To\_Hydrology, Horizontal\_Distance\_To\_Roadways , Horizontal\_Distance\_To\_Fire\_Points.

  
  
Negatively Skewed columns are-Elevation,Hillshade\_9am,Hillshade\_Noon,Hillshade\_3pm. The values of Skewness are also given above which shows how much particular feature is skewed.

### 4.6 SCATTERPLOT OF HILLSHADE INDEXES



The values of Hillshade Indexes as observed earlier also turnout to be in the range of 0 to 255. Feature Hillshade\_3pm contains many zeros. If there was shade at 9 AM and noon, it is very obvious to expect that there will be very little shade at 3 PM.

### 4.7 STATISTICAL ANALYSIS

#### 4.7.1 CHI SQUARE TEST

H0: the two samples are independent.  
H1: there is a dependency between the samples.

Also, we will be using Cramér's V Corrected Value as a measure to quantify the association among various categorical variables only.  
Cramér's V (sometimes referred to as Cramér's phi and denoted as φc) is a measure of association between two nominal variables, giving a value between 0 and +1 (inclusive). It is based on Pearson's chi-squared statistic

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Variable 1 | Variable 2 | Result | Cramer's Corrected Value |
| Chisquare Test of Association & Cramer's Corrected Value | Winderness\_types | Forest\_Cover\_Type | Dependency Found | 0.425 |
| Soil\_types | Forest\_Cover\_Type | Dependency Found | 0.607 |
| Winderness\_types | Soil\_types | Dependency Found | 0.685 |

Therefore, there is a dependency among all the three categorical Columns. According to the Cramér's V Corrected Value Wilderness\_types and Soil\_types are highly associated variables.

Reference: <https://en.wikipedia.org/wiki/Cram%C3%A9r%27s_V>

#### 4.7.2 ONE WAY ANOVA

Verifying whether all continuous variables are good for target variable predictions or not by performing One-way ANOVA among all Forest Cover Types

H0: the means of the samples are equal.

H1: one or more of the means of the samples are unequal.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Variable 1 | Variable 2 | Result |
| One-Way ANOVA | Elevation | Forest\_Cover\_Type\_designation | Unequal means |
| Aspect | Forest\_Cover\_Type\_designation | Unequal means |
| Slope | Forest\_Cover\_Type\_designation | Unequal means |
| Horizontal\_Distance\_To\_Hydrology | Forest\_Cover\_Type\_designation | Unequal means |
| Vertical\_Distance\_To\_Hydrology | Forest\_Cover\_Type\_designation | Unequal means |
| Horizontal\_Distance\_To\_Roadways | Forest\_Cover\_Type\_designation | Unequal means |
| Hillshade\_9am | Forest\_Cover\_Type\_designation | Unequal means |
| Hillshade\_Noon | Forest\_Cover\_Type\_designation | Unequal means |
| Hillshade\_3pm | Forest\_Cover\_Type\_designation | Unequal means |
| Horizontal\_Distance\_To\_Fire\_Points | Forest\_Cover\_Type\_designation | Unequal means |

Since, all the continuous variables are having one or more than one of the means among 4 forest cover types to be different from each other. So, in order to further understand the data we have to perform Post-Hoc Analysis.

#### 4.7.3 POST HOC ANALYSIS

For Elevation:

|  |
| --- |
| Multiple Comparison of Means - Tukey HSD, FWER=0.05 |
| ======================================================== |
| group1 group2 meandiff p-adj lower upper reject |
| -------------------------------------------------------- |
| 1 2 -207.7088 0.001 -208.9952 -206.4225 True |
| 1 3 -734.135 0.001 -736.6955 -731.5746 True |
| 1 7 233.2838 0.001 230.0089 236.5587 True |
| 2 3 -526.4262 0.001 -528.9396 -523.9128 True |
| 2 7 440.9926 0.001 437.7544 444.2309 True |
| 3 7 967.4188 0.001 963.4961 971.3415 True |
| -------------------------------------------------------- |

Thus, statistically all the combinations of Forest\_Cover\_Type\_designation are acting as distinct distributions w.r.t each other for Elevation.

For Aspect:

|  |
| --- |
| Multiple Comparison of Means - Tukey HSD, FWER=0.05 |
| ====================================================== |
| group1 group2 meandiff p-adj lower upper reject |
| ------------------------------------------------------ |
| 1 2 -4.0777 0.001 -4.8991 -3.2563 True |
| 1 3 20.2343 0.001 18.5992 21.8693 True |
| 1 7 -2.902 0.0021 -4.9932 -0.8108 True |
| 2 3 24.312 0.001 22.707 25.9169 True |
| 2 7 1.1757 0.4627 -0.8921 3.2435 False |
| 3 7 -23.1363 0.001 -25.6412 -20.6314 True |
| ------------------------------------------------------ |

Thus, statistically all the combinations except 2-7 of Forest\_Cover\_Type\_designation are acting as distinct distributions w.r.t each other for Aspect.

For Slope:

|  |
| --- |
| Multiple Comparison of Means - Tukey HSD, FWER=0.05 |
| =================================================== |
| group1 group2 meandiff p-adj lower upper reject |
| --------------------------------------------------- |
| 1 2 0.4234 0.001 0.3706 0.4761 True |
| 1 3 7.6431 0.001 7.5381 7.7481 True |
| 1 7 1.1288 0.001 0.9945 1.2631 True |
| 2 3 7.2197 0.001 7.1166 7.3228 True |
| 2 7 0.7054 0.001 0.5726 0.8382 True |
| 3 7 -6.5143 0.001 -6.6752 -6.3534 True |
| --------------------------------------------------- |

Thus, statistically all the combinations of Forest\_Cover\_Type\_designation are acting as distinct distributions w.r.t each other for Slope.

For Horizontal Distance To Roadways:

|  |
| --- |
| Multiple Comparison of Means - Tukey HSD, FWER=0.05 |
| =========================================================== |
| group1 group2 meandiff p-adj lower upper reject |
| ----------------------------------------------------------- |
| 1 2 -185.3037 0.001 -196.4615 -174.146 True |
| 1 3 -1670.8938 0.001 -1693.1035 -1648.684 True |
| 1 7 123.4159 0.001 95.009 151.8229 True |
| 2 3 -1485.5901 0.001 -1507.3915 -1463.7886 True |
| 2 7 308.7197 0.001 280.6308 336.8085 True |
| 3 7 1794.3097 0.001 1760.2838 1828.3356 True |
| ----------------------------------------------------------- |

Thus, statistically all the combinations of Forest\_Cover\_Type\_designation are acting as distinct distributions w.r.t each other for Horizontal\_Distance\_To\_Roadways.

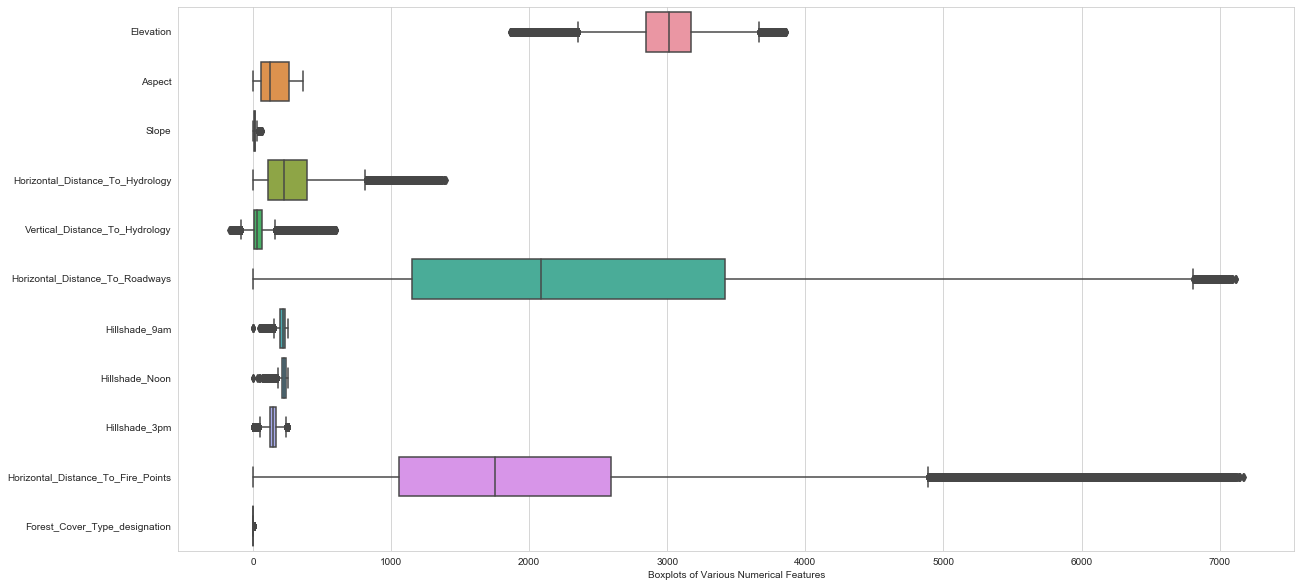
For Hillshade\_Noon:

|  |
| --- |
| Multiple Comparison of Means - Tukey HSD, FWER=0.05 |
| =================================================== |
| group1 group2 meandiff p-adj lower upper reject |
| --------------------------------------------------- |
| 1 2 1.8964 0.001 1.7548 2.038 True |
| 1 3 -7.6037 0.001 -7.8855 -7.3219 True |
| 1 7 -1.6842 0.001 -2.0446 -1.3237 True |
| 2 3 -9.5001 0.001 -9.7767 -9.2234 True |
| 2 7 -3.5806 0.001 -3.937 -3.2242 True |
| 3 7 5.9195 0.001 5.4878 6.3512 True |
| --------------------------------------------------- |
|  |

Thus, statistically all the combinations of Forest\_Cover\_Type\_designation are acting as a distinct distribution w.r.t each other for Hillshade\_3pm, Hillshade\_9am & Hillshade\_Noon as all these features also tend to show similar results.

# 5. DATA CLEANING

### 5.1 TRANSFORMATION



Since the data is having a lot of outliers and also we can do outlier treatment because the nature of the data is not critical in nature, outlier treatment is advisable along with Sensitivity analysis for experimental purposes in order to understand which combination is best suitable for our analysis.

We will be applying transformations, feature engineering, binning through quantile cut and winsorization technique upon various features. Also, we will consider understanding the underlying mathematics behind Hill Shading Techniques.

In order to understand what better features can be engineered on the basis of appropriate domain knowledge.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Transformations v/s Model Errors** | **Naïve Bayes** | | **Decision Tree** | | **Bagged Decision Tree** | | **Random Forest** | | **Boosted Decision Tree** | | **Bagged RG Decision Tree** | |
| **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** |
|  |  |  |  |  |  |  |  |  |  |  |  |
| **MinMax Scaler + Stratified K-Fold CV** | 0.893098 | 0.000001 | 0.05112 | 0.000001 | 0.031982 | 0.000001 | 0.044753 | 0.000001 | 0.050997 | 0.000001 | 0.225795 | 0.000008 |
| **Standard Scaler + Stratified K-Fold CV** | 0.89376 | 0.000001 | 0.051142 | 0.000001 | 0.032001 | 0.000001 | 0.04466 | 0.000001 | 0.050984 | 0.000001 | 0.225786 | 0.000008 |
| **MinMax Scaler + K-Fold CV** | 0.893396 | 0.000001 | 0.050856 | 0 | 0.032333 | 0 | 0.044287 | 0.000002 | 0.050925 | 0 | 0.22711 | 0.000005 |
| **Standard Scaler + K-Fold CV** | 0.893758 | 0.000001 | 0.050889 | 0 | 0.032321 | 0 | 0.044294 | 0.000002 | 0.050956 | 0 | 0.227109 | 0.000006 |
| **Power Transformer + K-Fold CV** | 0.893758 | 0.000001 | 0.050866 | 0 | 0.032328 | 0 | 0.044407 | 0.000002 | 0.050989 | 0 | 0.227111 | 0.000005 |
| **Power Transformer + Stratified K-Fold CV** | 0.89376 | 0.000001 | 0.051158 | 0.000001 | 0.031959 | 0.000001 | 0.044682 | 0.000001 | 0.050984 | 0.000001 | 0.22579 | 0.000008 |

Thus, after getting the best Results with Min-Max Scaler along with Stratified K-Fold Cross Validation. We will go for the MinMaxScaler Transformation in the Future Models along with Stratified K-Fold for best results.

### 5.2 OUTLIER TREATMENTS

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **OT Attempts v/s Model Errors** | **Naïve Bayes** | | **Decision Tree** | | **Bagged Decision Tree** | | **Random Forest** | | **Boosted Decision Tree** | | **Bagged RG Decision Tree** | |
| **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** |
|  |  |  |  |  |  |  |  |  |  |  |  |
| **After applying log-transform on Elevation** | 0.892631 | 0.000001 | 0.051104 | 0.000001 | 0.051104 | 0.000001 | 0.044793 | 0.000001 | 0.050995 | 0.000001 | 0.225795 | 0.000008 |
| **After applying pd-qcut on Elevation** | 0.74339 | 0.000211 | 0.055797 | 0 | 0.037779 | 0 | 0.051335 | 0 | 0.0562 | 0 | 0.240116 | 0.000003 |

Looking at the above observations which are obtained after applying different outlier treatment techniques, we have recorded variance errors and bias errors and compared these recorded values with the base model errors.

All those values which are in green color are those which are better than the base model errors. Through this we can say that applying log transformation on the Elevation seems to give us reduced biased errors in Naive Bayes ,Decision Tree and in Boosted Decision Tree model as observed above.

# 

# 6. INITIAL APPROACH

1. Made a Decision Tree model on the entire data to see its overall behaviour and check entropy.

|  |
| --- |
| precision recall f1-score support |
|  |
| 1 0.94 0.94 0.94 42393 |
| 2 0.95 0.95 0.95 56673 |
| 3 0.98 0.97 0.98 7092 |
| 7 0.95 0.95 0.95 4123 |
|  |
| accuracy 0.95 110281 |
| macro avg 0.95 0.95 0.95 110281 |
| weighted avg 0.95 0.95 0.95 110281 |

While the accuracy of this simple Decision Tree is high, and also precision and recall is also high for each of the predicted classes. but as we know, decision tree has a natural tendency to overfit thereby we will be studying combinations of different tree based models.

# 

# 

# 7. SENSITIVITY ANALYSIS

Since we are dealing with cartographic data, we will be focusing more on F1-score rather than Recall or Precision i.e, we will allow a balance between both Type II error & Type I error in our models because in our case, we are having a multi-class problem. Thus, allowing FP or FN w.r.t any one class will create an increase in FN in the other class which we don’t want.

Understandably, if we would have had just two classes then we could have chosen Precision over Recall but in this multiclass problem a weighted f1-score should be the best metric to come up with the best results.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **FE Attempts v/s Model Errors** | **Naïve Bayes** | | **Decision Tree** | | **Bagged Decision Tree** | | **Random Forest** | | **Boosted Decision Tree** | | **Bagged RG Decision Tree** | |
| **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** | **Bias Error** | **Variance Error** |
|  |  |  |  |  |  |  |  |  |  |  |  |
| **After Dropping Soil Types Feature** | 0.723627 | 0 | 0.061001 | 0.000001 | 0.036646 | 0.000001 | 0.045877 | 0.000001 | 0.060924 | 0.000001 | 0.234422 | 0.000002 |
| **After FE of Shortest Distance to Hydrology** | 0.893091 | 0.000001 | 0.051961 | 0.000001 | 0.032215 | 0.000001 | 0.04458 | 0.000001 | 0.051911 | 0.000001 | 0.22599 | 0.000009 |
| **After FE of Hillshade mean** | 0.893176 | 0 | 0.051589 | 0.000001 | 0.03251 | 0 | 0.047703 | 0 | 0.051596 | 0.000001 | 0.225798 | 0.000008 |
| **After FE of Cos-Sine of Slope** | 0.889699 | 0.000002 | 0.051139 | 0.000001 | 0.032098 | 0.000001 | 0.053805 | 0 | 0.0512 | 0.000001 | 0.225807 | 0.000008 |
| **After FE Sum & Difference of Distances** | 0.888571 | 0.000002 | 0.0431 | 0 | 0.026166 | 0.000001 | 0.025104 | 0.000001 | 0.043275 | 0.000001 | 0.225968 | 0.000015 |
| **After FE Sum, Diff & Multipli of Hillshade Index** | 0.893051 | 0.000001 | 0.054367 | 0.000001 | 0.033735 | 0.000001 | 0.069138 | 0.000005 | 0.053692 | 0 | 0.226651 | 0.000004 |
| **After Binning Soil Types** | 0.631154 | 0 | 0.058034 | 0.000001 | 0.034942 | 0.000001 | 0.046167 | 0.000001 | 0.057824 | 0.000001 | 0.233892 | 0.000006 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

INFERENCE: In this case we have observed the highest decrease in errors; but we have neither precision nor recall. The model is not able to identify the forest cover types at an effective rate in case of Naive Bayes modelling technique. After creating a Feature engineering with the sums and differences among the horizontal distances. We have found that most of the models except Naive Bayes have shown considerable improvement. Thus, we will be taking this forward in further experiments as well.

# 8. APPENDIX

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Measurement | Description |
| Elevation | Quantitative | meters | Elevation in meters |
| Aspect | Quantitative | azimuth | Aspect in degrees azimuth |
| Slope | Quantitative | degrees | Slope in degrees |
| Horizontal\_Distance\_To\_Hydrology | Quantitative | meters | Horz Dist to nearest surface water features |
| Vertical\_Distance\_To\_Hydrology | Quantitative | meters | Vert Dist to nearest surface water features |
| Horizontal\_Distance\_To\_Roadways | Quantitative | meters | Horz Dist to nearest roadway |
| Hillshade\_9am | Quantitative | 0 to 255 index | Hillshade index at 9am, summer solstice |
| Hillshade\_Noon | Quantitative | 0 to 255 index | Hillshade index at noon, summer solstice |
| Hillshade\_3pm | Quantitative | 0 to 255 index | Hillshade index at 3pm, summer solstice |
| Horizontal\_Distance\_To\_Fire\_Points | Quantitative | meters | Horz Dist to nearest wildfire ignition points |
| Wilderness\_Area (4 binary columns) | Quantitative | 0 (absence) or 1 (presence) | Wilderness area designation |
| Soil\_Type (40 binary columns) | Quantitative | 0 (absence) or 1 (presence) | Soil Type designation |
| Cover\_Type (7 types) | Integer | 1 to 7 | Forest Cover Type designation |

### 8.1 BASE MODEL’S CODE

Using Decision Tree:

dummy\_df = pd.get\_dummies(df, drop\_first=True)

X = dummy\_df.drop(columns='Forest\_Cover\_Type\_designation')

y = dummy\_df['Forest\_Cover\_Type\_designation']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size= 0.2, random\_state= 123)

sc = StandardScaler()

X\_train\_std = sc.fit\_transform(X\_train)

X\_test\_std = sc.transform(X\_test)

dt = DecisionTreeClassifier(random\_state = 0)

model\_1 = dt.fit(X\_train\_std, y\_train)

y\_pred\_1 = model\_1.predict(X\_test\_std)

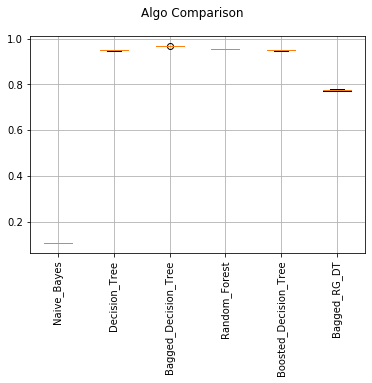
print(classification\_report(y\_test, y\_pred\_1))

print('Training Score',model\_1.score(X\_train\_std, y\_train))

print('Testing Score',model\_1.score(X\_test\_std, y\_test))

Final Base Modelling Report

modelis(X, method = 'minmax', fold= 'skf')



Results:

Naive\_Bayes: 0.893098, (0.000001)  
Decision\_Tree: 0.051120, (0.000001)  
Bagged\_Decision\_Tree: 0.031982, (0.000001)  
Random\_Forest: 0.044753, (0.000001)  
Boosted\_Decision\_Tree: 0.050997, (0.000001)  
Bagged\_RG\_DT: 0.225795, (0.000008)

### 

### 8.2 FUNCTION USED FOR MODELING def modelis(X, method, fold):

method = str(method)

if method == 'minmax':

mn = MinMaxScaler()

X\_std = mn.fit\_transform(X)

elif method == 'power':

pt = PowerTransformer()

X\_std = pt.fit\_transform(X)

else:

sc = StandardScaler()

X\_std = sc.fit\_transform(X)

#gb = GradientBoostingClassifier()

nb = GaussianNB()

dt = DecisionTreeClassifier(criterion = 'gini', random\_state=0) # Specifying random state is to 0

rg\_dt = DecisionTreeClassifier(criterion = 'gini', random\_state=0, max\_depth=7)

rf = RandomForestClassifier(n\_estimators= 10, criterion= 'entropy', random\_state=0)

dt\_bag = BaggingClassifier(base\_estimator = dt, n\_jobs=-1, random\_state=0)

dt\_rg\_bag = BaggingClassifier(base\_estimator = rg\_dt, n\_jobs=-1, random\_state=0)

dt\_boost = AdaBoostClassifier(base\_estimator= dt, n\_estimators= 10, random\_state=0)

#rf\_boost = AdaBoostClassifier(base\_estimator= rf, n\_estimators= 100, random\_state=0)

models = []

#models.append(('Gradient\_Boosting', gb))

models.append(('Naive\_Bayes', nb))

models.append(('Decision\_Tree', dt))

models.append(('Bagged\_Decision\_Tree',dt\_bag))

models.append(('Random\_Forest', rf))

models.append(('Boosted\_Decision\_Tree', dt\_boost))

models.append(('Bagged\_RG\_DT', dt\_rg\_bag))

result= []

names = []

for name,model in models:

if fold == 'kf':

skf = KFold(shuffle=True, n\_splits=5, random\_state=0)

elif fold == 'skf':

skf = StratifiedKFold(shuffle=True, n\_splits=5, random\_state=0)

else:

skf = KFold(shuffle=True, n\_splits=5, random\_state=0)

cv\_results = cross\_val\_score(model, X\_std, y, cv=skf, scoring='f1\_weighted')

result.append(cv\_results)

names.append(name)

print('%s: %f, (%f)' %(name, 1-np.mean(cv\_results), np.var(cv\_results, ddof=1)))

fig = plt.figure()

plt.grid()

fig.suptitle('Algo Comparison')

ax = fig.add\_subplot(111)

plt.xticks(rotation=90)

plt.boxplot(result)

ax.set\_xticklabels(names)

plt.show()

### 8.3 MODELING AFTER DROPPING SOIL\_TYPES

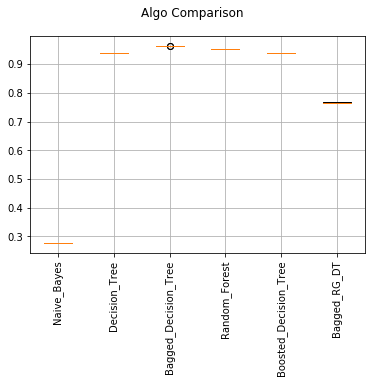
df\_w\_soil = df.drop(columns = 'Soil\_types')

dummy\_w\_soil = pd.get\_dummies(df\_w\_soil, drop\_first=True)

X = dummy\_w\_soil.drop(columns='Forest\_Cover\_Type\_designation')

y = dummy\_w\_soil['Forest\_Cover\_Type\_designation']

modelis(X, sc)



Results-

Naive\_Bayes: 0.723627, (0.000000)  
Decision\_Tree: 0.061001, (0.000001)  
Bagged\_Decision\_Tree: 0.036646, (0.000001)  
Random\_Forest: 0.045877, (0.000001)  
Boosted\_Decision\_Tree: 0.060924, (0.000001)

Bagged\_RG\_DT: 0.234422, (0.000002)

### 8.4 FUNCTION THAT USES MODELIS

def modelis2(df\_feat, method, fold):

dummy\_feat = pd.get\_dummies(df\_feat, drop\_first=True)

X = dummy\_feat.drop(columns='Forest\_Cover\_Type\_designation')

y = dummy\_feat['Forest\_Cover\_Type\_designation']

modelis(X, method, fold)

### 8.5 FUNCTION TO OUTPUT RANDOM FOREST FEATURE IMPORTANCE

def rf\_imp(df\_add\_subt,i):

dummy\_feat = pd.get\_dummies(df\_add\_subt, drop\_first=True)

X = dummy\_feat.drop(columns=['Forest\_Cover\_Type\_designation'])

y = dummy\_feat['Forest\_Cover\_Type\_designation']

sc= StandardScaler()

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.2, random\_state = 123)

X\_train\_std = pd.DataFrame(sc.fit\_transform(X\_train), columns = X\_train.columns)

X\_test\_std = pd.DataFrame(sc.transform(X\_test), columns = X\_test.columns)

rf = RandomForestClassifier(random\_state = 0)

model\_1 = rf.fit(X\_train\_std, y\_train)

pd.DataFrame(rf.feature\_importances\_, index = X\_train.columns)[0].sort\_values(ascending=False)[:i].plot(kind='barh')

### 8.6 MODEL WITH FE SHORTEST DISTANCE TO HYDROLOGY

df\_feat = df.copy()

df\_feat['Shortest\_Distance\_to\_Hydrology'] = np.sqrt(df['Horizontal\_Distance\_To\_Hydrology']\*\*2 + df['Vertical\_Distance\_To\_Hydrology']\*\*2)

modelis2(df\_feat, method = 'minmax', fold= 'skf')



Results-

Naive\_Bayes: 0.893091, (0.000001)

Decision\_Tree: 0.051961, (0.000001)

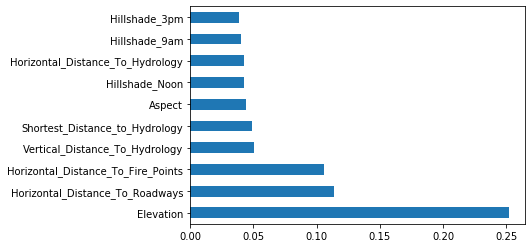
Bagged\_Decision\_Tree: 0.032215, (0.000001)

Random\_Forest: 0.044580, (0.000001)

Boosted\_Decision\_Tree: 0.051911, (0.000001)

Bagged\_RG\_DT: 0.225990, (0.000009)

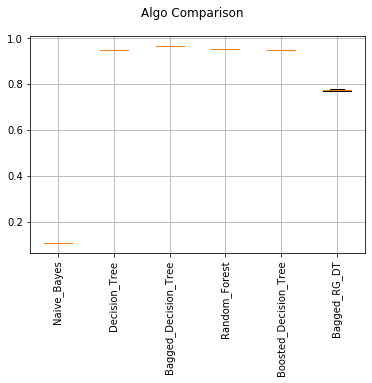
rf\_imp(df\_feat,10)



### 8.7 MODEL WITH FE HILLSHADE INDEX MEAN

df\_hill\_mean = df.copy()

df\_hill\_mean['Hillshade\_mean'] = round((df['Hillshade\_9am'] + df['Hillshade\_Noon'] + df['Hillshade\_3pm'])/3,2)

modelis2(df\_hill\_mean, method = 'minmax', fold= 'skf')  


Results-

Naive\_Bayes: 0.893176, (0.000000)

Decision\_Tree: 0.051589, (0.000001)

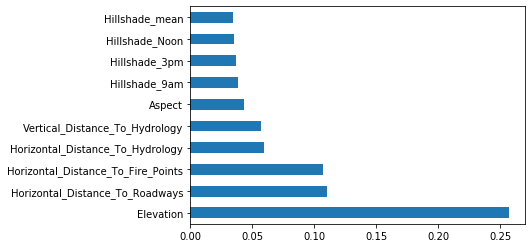
Bagged\_Decision\_Tree: 0.032510, (0.000000)

Random\_Forest: 0.047703, (0.000000)

Boosted\_Decision\_Tree: 0.051596, (0.000001)

Bagged\_RG\_DT: 0.225798, (0.000008)

rf\_imp(df\_hill\_mean,10)



### 8.8 MODEL WITH SIN-COS OF SLOPE

df\_sin\_cos = df.copy()

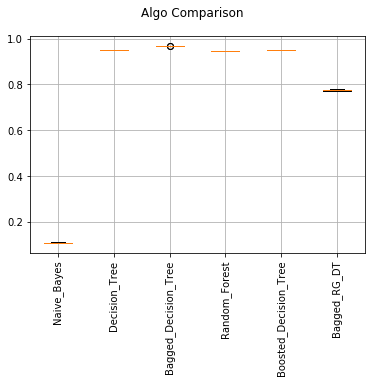
import math

df\_sin\_cos['Slope\_rad'] = (math.pi/180)\*(df\_sin\_cos['Slope'])

df\_sin\_cos['Slope\_rad\_sin'] = np.sin(df\_sin\_cos['Slope\_rad'])

df\_sin\_cos['Slope\_rad\_cos'] = np.cos(df\_sin\_cos['Slope\_rad'])

modelis2(df\_sin\_cos, method = 'minmax', fold= 'skf')



Results-

Naive\_Bayes: 0.889699, (0.000002)

Decision\_Tree: 0.051139, (0.000001)

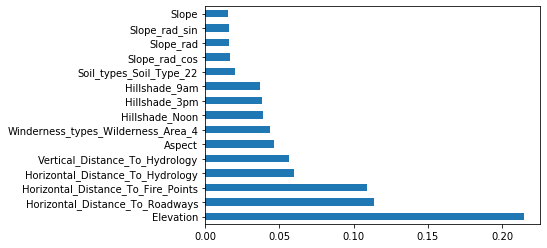
Bagged\_Decision\_Tree: 0.032098, (0.000001)

Random\_Forest: 0.053805, (0.000000)

Boosted\_Decision\_Tree: 0.051200, (0.000001)

Bagged\_RG\_DT: 0.225807, (0.000008)

rf\_imp(df\_sin\_cos,15)



### 8.9 MODEL WITH ADDING-SUBTRACTING ALL COMBINATIONS OF HORIZONTAL DISTANCES

df\_add\_subt = df.copy()

df\_add\_subt['Road\_p\_Fire'] = df\_add\_subt['Horizontal\_Distance\_To\_Roadways'] + df\_add\_subt['Horizontal\_Distance\_To\_Fire\_Points']

df\_add\_subt['Road\_s\_Fire'] = df\_add\_subt['Horizontal\_Distance\_To\_Roadways'] - df\_add\_subt['Horizontal\_Distance\_To\_Fire\_Points']

df\_add\_subt['Hyd\_s\_Fire'] = df\_add\_subt['Horizontal\_Distance\_To\_Hydrology'] - df\_add\_subt['Horizontal\_Distance\_To\_Fire\_Points']

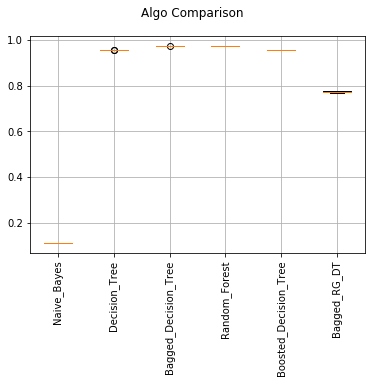
df\_add\_subt['Hyd\_p\_Fire'] = df\_add\_subt['Horizontal\_Distance\_To\_Hydrology'] + df\_add\_subt['Horizontal\_Distance\_To\_Fire\_Points']

df\_add\_subt['Hyd\_p\_Road'] = df\_add\_subt['Horizontal\_Distance\_To\_Hydrology'] + df\_add\_subt['Horizontal\_Distance\_To\_Roadways']

df\_add\_subt['Hyd\_s\_Road'] = df\_add\_subt['Horizontal\_Distance\_To\_Hydrology'] - df\_add\_subt['Horizontal\_Distance\_To\_Roadways']

df\_add\_subt['Shortest\_Distance\_to\_Hydrology'] = np.sqrt(df['Horizontal\_Distance\_To\_Hydrology']\*\*2 + df['Vertical\_Distance\_To\_Hydrology']\*\*2)

modelis2(df\_add\_subt, method = 'minmax', fold= 'skf')



Results-

Naive\_Bayes: 0.888571, (0.000002)

Decision\_Tree: 0.043100, (0.000000)

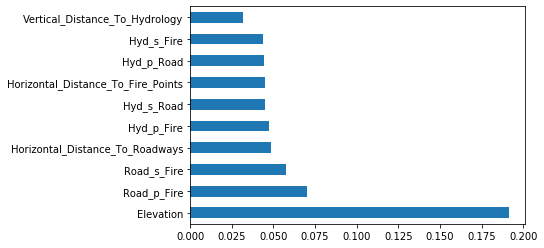
Bagged\_Decision\_Tree: 0.026166, (0.000001)

Random\_Forest: 0.025104, (0.000001)

Boosted\_Decision\_Tree: 0.043275, (0.000001)

Bagged\_RG\_DT: 0.225968, (0.000015)

rf\_imp(df\_add\_subt,10)

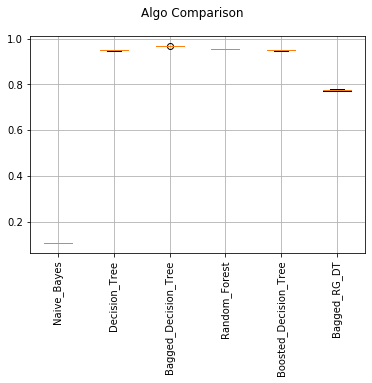


### 

### 8.10 AFTER APPLYING LOG-TRANSFORM ON ELEVATION

trans\_data.Elevation=np.log(trans\_data.Elevation)

modelis2(trans\_data, method = 'minmax', fold= 'skf')

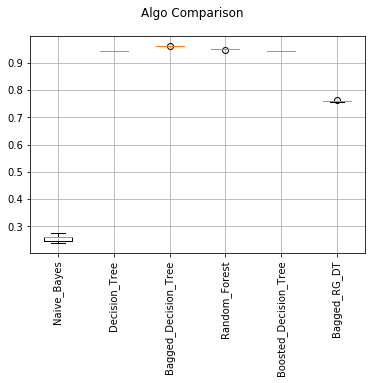


RESULTS-

Naive\_Bayes: 0.892631, (0.000001)  
Decision\_Tree: 0.051104, (0.000001)  
Bagged\_Decision\_Tree: 0.031981, (0.000001)  
Random\_Forest: 0.044793, (0.000001)  
Boosted\_Decision\_Tree: 0.050995, (0.000001)

Bagged\_RG\_DT: 0.225795, (0.000008)

### 8.11 AFTER APPLYING PD-QCUT ON ELEVATION trans\_data=new\_data.copy() trans\_data.Elevation=pd.qcut(trans\_data.Elevation,q=7) modelis2(trans\_data, method = 'minmax', fold= 'skf')



RESULTS-

Naive\_Bayes: 0.743390, (0.000211)  
Decision\_Tree: 0.055797, (0.000000)  
Bagged\_Decision\_Tree: 0.037779, (0.000000)  
Random\_Forest: 0.051335, (0.000000)  
Boosted\_Decision\_Tree: 0.056200, (0.000000)

Bagged\_RG\_DT: 0.240116, (0.000003)